A COMPUTER ALGORITHM TO OPTIMIZE THE SCHEDULING OF STRATEGIC SEALIFT

A Thesis

by

GARRETT RANDALL LAMBERT

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 1995

Major Subject: Industrial Engineering

A COMPUTER ALGORITHM TO OPTIMIZE THE SCHEDULING OF STRATEGIC SEALIFT

A Thesis

by

GARRETT RANDALL LAMBERT

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Accesion For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

May 1995

Major Subject: Industrial Engineering

ABSTRACT

A Computer Algorithm to Optimize the Scheduling

of Strategic Sealift. (May 1995)

Garrett Randall Lambert, B.S., United States Military Academy

Chair of Advisory Committee: Dr. Guy L. Curry

The problem of scheduling strategic sealift assets for a U.S. Army deployment in response to a major regional contingency is considered in this paper. The complexity of this problem depends upon the different types of ship speeds and capacities as well as unit precedence constraints. The objective is to minimize the sum of weighted unit tardiness. This paper presents the development and solution of a realistic strategic sealift scheduling problem based upon the experiences gained during Operation Desert Shield. A mathematical model for the problem is proposed and an algorithm is developed and applied to solve the scheduling problem. Results of the algorithm are compared with randomly generated schedules to determine algorithm effectiveness.

TABLE OF CONTENTS

LIST OF TABLES

INTRODUCTION

Recent changes in the post cold war world accentuate the need for a sufficient and timely deployment of heavy combat forces to various regions of the world. Before the dissolution of the Soviet Union, a key component of United States military strategy was forward-stationed forces. Today, fewer U.S. forces are stationed overseas due to the reduced Soviet threat and treaty obligations. This reduction in forward-deployed forces has caused a new military strategy to be developed which depends on rapid deployment and force projection to protect vital U.S. interests (Willie 1992).

The transportation of U.S. forces from "fort to foxhole" is accomplished through the integration of three components: airlift, sealift, and pre-positioned equipment. The focus of this study will be on the second element of this triad, namely strategic sealift. Any contingency requiring heavy combat forces will depend greatly upon sealift assets to deploy the required combat power to the zone of conflict.

The Iraqi invasion of Kuwait on 2 August 1990 resulted in the largest U.S. Military deployment since the Vietnam War (U.S. Congress 1991). This crisis, known as Operation Desert Shield/Desert Storm, severely tested the ability of the United States to project combat power to a distant theater of operations and therefore serves as a good model for determining future transportation requirements. Sealift in Operation Desert Shield/Desert Storm accounted for about 95% of all cargo transported (Willie 1992). This fact underscores the tremendous importance of strategic sealift assets to U.S. strategic mobility.

This thesis follows the format of the *Operations Research* journal.

The U.S. strategic sealift capability is made up of ships from three separate ship forces: the Ready Reserve Force (RRF), the Military Sealift Command's (MSC) fast sealift ships (FSS), and the afloat pre-positioning force (APF). These are the most responsive and militarily useful ships available. Additional sources of ships are U.S.-flag commercial ships, foreign-flag commercial vessels, and the Effective U.S.-Controlled Fleet (EUSC) which are U.S.-owned foreign-flagged vessels (U.S. Congress 1991). These other sources, however, are less responsive and have few militarily useful ships.

The high cost of obtaining and maintaining these sealift assets and the high stakes endemic to the projection of U.S. forces in support of national objectives highlights the importance of proper utilization of sealift assets. One of the critical aspects of strategic sealift then is to optimize the scheduling of sealift assets to ensure that the required number of combat forces is deployed to a theater in the minimum possible time. This scheduling problem involves a set of demands represented by the different army units at different ports which need to be deployed to a destination port in the theater of operations. Units are arranged in priority order with the unit having the higher priority being satisfied before a unit with a lower priority. The ships represent the resources which service the demand. The ships that make up the available sealift assets have many different characteristics which affect the rate at which they can satisfy demand. These are ship capacity, speed, readiness posture, initial location, and load/unload characteristics.

This research is motivated by an interest in the impact of the strategic sealift scheduling problem on future U.S. strategic mobility requirements. It has led to the

development and application of a solution procedure to the issue of scheduling strategic sealift for U.S. Army forces. The main objective of this research is to develop an algorithm to solve a specific strategic sealift scheduling problem and to provide a computer program for the solution procedure. The algorithm will be tested against a real-world problem based upon the experiences of Desert Shield. The final product of this research is a decision-support methodology that can be used to determine the best schedule for a set of ships given the ship speeds, initial locations, and availability dates, the distance between ports, the demands at each port, and the desired unit sequence of deployment.

LITERATURE REVIEW

A literature review was conducted to determine if any previous work had addressed or solved the strategic sealift scheduling problem. The following is a summary of that review.

Ronen (1983) conducted a survey of cargo ship routing and scheduling problems. In it he suggests a classification scheme for ship routing and scheduling problems and models. His survey of the work done in ship scheduling illustrates the wide variety of ship routing and scheduling problems. Several papers in his survey contains aspects of the problem outlined in this proposal but none addressed them all. Most of the work surveyed did not take into account the different speeds of ships in a fleet or made other simplifying assumptions.

Laderman, Glieberman, and Egan (1966) present a linear programming model of a fleet of vessels with different characteristics which are required to transport quantities of cargo from producing ports to destination ports over a given planning horizon. The objective was to minimize the number of ships required and to allocate the resulting required ships to routes. Non-integer number of routes were rounded. Precedence constraints on cargoes were not included in the model.

Bellmore, Bennington, and Lubore (1971) presented a mixed integer linear program for a multivehicle tanker scheduling problem with time windows. Their formulation allowed for different carrying capabilities and partial loading of the tankers from a fixed fleet. Their objective was to determine a schedule and a routing for the fleet with maximum utility. They propose using the Dantzig-Wolfe decomposition technique

coupled with a branch and bound algorithm to solve the problem but do not provide any results.

McKay and Hartley (1974) extended the tanker scheduling problem above by allowing multiple loading ports and adding port constraints such as draft restrictions. Their objective was to minimize operating costs of the tankers and costs of purchasing products at loading ports. They provide an integer programming formulation and restructure this formulation by restricting the set of possible routes to a much smaller set of "acceptable" routes. An approximate solution technique to this formulation was proposed by the relaxation of integer restrictions, solving the resulting linear program, and then using a rounding procedure for the integer variables. An optimal solution is not guaranteed using this procedure.

Fisher and Rosenwein (1989) present an algorithm they developed for the Military Sealift Command (MSC) for scheduling a fleet of tankers carrying bulk cargoes. They allow for ships with different speeds and capacities, port constraints, time-window constraints for cargo, and operating costs for the fleet. Their objective is to minimize operating costs of the fleet. This work is very closely related to the strategic sealift scheduling problem. Their solution technique hinges on the generation of candidate schedules for each ship. Once this set of candidate schedules is generated, the problem is then formulated as a set-packing problem and solved using a branch and bound procedure with bounds obtained by applying Lagrangian relaxation. The details of this algorithm were not presented. Computational experience with real data from MSC was presented indicating substantial savings over manual scheduling procedures.

This review of the literature has not found any solution to the problem to be addressed in this thesis short of complete enumeration. Some of the methods above, particularly the approach by Fisher and Rosenwein (1989), provided insight into possible solution approaches to this problem.

METHODOLOGY

Introduction

The methodology used to solve this problem consisted of the following:

(1) Problem Description. An in depth study of U.S. strategic sealift policy was conducted to better understand the problem domain. Strategic sealift operations during Desert Shield were studied to assist in the development of a realistic problem set.

(2) Problem Formulation. A mathematical model was developed based upon the objectives, constraints, and assumptions of the problem.

(3) Solution Procedure. A heuristic algorithm was developed and applied to obtain a solution to the strategic sealift scheduling problem.

The next three sections provide a detailed description of the problem, the formulation of the problem, and the solution procedure used to solve the problem.

Problem Description and Assumptions

In order to formulate and solve the strategic sealift problem, an understanding of the problem domain as well as the methodology used to define the specific problem set is necessary. What follows is a discussion of the key issues concerning the problem of scheduling strategic sealift. This discussion includes the major elements of the strategic sealift scheduling problem, how the data set defining the problem (Data Set I) was developed, and the major assumptions made regarding the problem.

Schank et al. (1991) refer to crisis action planning as one of the three types of strategic mobility planning. Crisis action planning refers to planning that occurs immediately

before and during a real conflict. Crisis action planning can be divided into two categories: (1) surge shipping during initial mobilization , and (2) resupply (sustainment) shipping to sustain the deployed forces. During the initial surge phase, shipping requirements consist mainly of outsize cargo such as tanks, helicopters, and other bulky military vehicles and unit equipment (U.S. Congress 1990). During Operation Desert Shield, the initial surge phase lasted roughly 90 days. During this period, about three and one-third Army divisions along with their associated headquarters, support equipment, and supplies were deployed to Saudi Arabia (U.S. Congress 1991). The second category, sustainment shipping, occurs throughout the deployment to sustain the force and to build up reserve stocks. This research will focus on the problem of scheduling sealift during the surge phase of crisis action planning. The Desert Shield experience during the first 90 days will be used as a guide for defining the problem to be solved and for generating a realistic problem data set.

The key components of the problem consist of the units to be deployed, the ports from and to which each unit will deploy, the ships that will transport them, and the routes the ships will traverse. The critical information necessary to characterize the scheduling problem during the surge phase can be assumed to be known or deterministic. The following is a brief description of each of the critical elements of the problem.

A unit is defined by its initial location, seaport of embarkation (SPOE), cargo requirements, priority of movement (precedence), date ready to load at the SPOE, seaport of debarkation (SPOD), and due date. The particular units chosen for this problem are the same units which were deployed during the surge phase of Operation

Desert Shield. The designated units and their key characteristics are shown in Table 1 below. A detailed listing of unit data is located in Appendix A.

Table 1. Unit Characteristics

Unit	Cargo(sq ft)	Priority	SPOE	SPOD
24th MECH	2,375,000	1	Savannah,GA	Ad Dammam, SA
101st AASLT	406,250	2	Mobile, AL	Ad Dammam, SA
3rd ACR	770,000	3	Beaumont, TX	Ad Dammam, SA
1st CAV	2,415,000	4	Beaumont, TX	Ad Dammam, SA
COSCOM	4,033,750	Multiple	Wilmington,NC/ Beaumont,TX	Ad Dammam, SA
Total	10,000,000			

All units are assumed to be available at their respective SPOEs at the start of the problem. Thus each unit is available and ready to load at its SPOE at time equals zero. The selected unit SPOEs for this problem are also shown in Table 1. The SPOD selected for all units is Ad Dammam port in Saudi Arabia. Al Jubayl port was also used during Desert Shield but it is close enough to Ad Dammam (within 30NM) for the two ports to be considered as one port (Defense Mapping Agency 1985).

The cargo requirements for each unit are dependent upon the type of unit. For example, an armored division (which consists of about 300 tanks) will require a great deal more transportation assets than an air assault division (which consists primarily of light infantry forces). Unit cargo requirements were based upon the planning factors given in the Deployment Planning Guide provided by the U.S. Army Military Traffic Management Command Transportation Engineering Agency (1991). Cargo for sealift can be measured in weight (short tons), volume (cubic feet) or area (square feet)

depending upon the type of cargo. For the heavy unit equipment characteristic of the surge phase, the area tends to be the constraining factor for ship loading (U.S. Congress 1991). Therefore cargo requirements for the units are expressed in square feet as shown in Table 1. A total of 10 million square feet of unit equipment is to be moved for this problem.

Unit priorities also reflect the Desert Shield experience and were obtained from U.S. Congress (1991). Unit priorities permit a general articulation of unit precedence for sealift assets and serve as a guide for the flow of the major units. Within a unit, such as the Corps Support Command (COSCOM), there may be several different priorities. Varying priorities within a unit can be further refined by assigning within-unit precedence constraints called unit types. For example, several sub-units in the COSCOM with the same priority can be further ordered by assigning them type numbers accordingly.

Ports have several characteristics which may influence the problem. A port's staging capacity is a measure of the allowable space for parking unit equipment prior to or after loading. Thus, insufficient staging space for all the units leaving or entering a port imposes a constraint upon the number of units which can occupy a port at any given time. A second constraint is the number of berths available in the port. This restricts the number of ships which can be loading/unloading in a port at any given time. Finally, the channel depth of a port can restrict the type of ships which may enter the port due to a ship's draft characteristics. This particular restriction may force equipment to be transloaded via barges so that it may be offloaded in the port. The ports used for this problem will be assumed to be unconstrained for the factors listed above. This turns out

to be a safe assumption for this particular problem. Since there are four SPOEs, the number of ships berthed at any one of these ports at one time will be less than the number possible at the single SPOD. Thus the SPOD will most likely have the larger number of ships/units to deal with at any given time since it is the focal point for all the units. The Desert Shield deployment required the ability to offload 10 to 12 ships simultaneously in Saudi Arabia during the peak periods of the deployment (U.S. Congress 1991). The Ad Dammam and Al Jubayl ports together have 56 berths and over 42 million square feet of storage space. Therefore the assumption of unconstrained port capacity is not completely unrealistic here. A summary of the distances between the ports relevant to the problem is contained in Table 2 below.

Table 2. Distances Between Ports (Nautical Miles)

	UNIT PORTS				
	SPOE1	SPOE2	SPOE3	SPOE4	SPOD
SHIP PORTS	WILM	SVNH	MOBL	BMNT	DAMM
WILM	0	132	1,206	1,436	8,623
SVNH	132	0	1,074	1,304	9,175
MOBL	1,206	1,074	0	230	9,580
BMNT	1,436	1,304	230	0	9,810
DAMM	8,623	9,175	9,580	9,810	0
JRRF	245	377	1,451	1,681	N/A
JACK	212	80	994	1,224	N/A
SPDO	4,695	4,563	4,326	4,473	N/A
SFRN	5,027	4,895	4,658	4,805	N/A
PTLD	5,451	5,319	5,082	5,229	N/A
DGCA	N/A	N/A	N/A	N/A	2,932

There are a number of ship characteristics which have an impact on the strategic sealift scheduling problem. These include: source of shipping, readiness, type of ship, crew availability, ship speed, load/unload times, initial location, and maintenance profile. The ships employed during the initial phase of the Desert Shield deployment came from a wide variety of sources but the brunt of the work (about 70%) was accomplished by U.S. government-controlled ships (Hura 1993). Commercial U.S.-flag and foreign-flag ships contributed to the surge phase deployment to a smaller degree than the U.S. government- controlled ships and therefore will not be considered in this problem. The U.S. Government-controlled ships consist of the Fast Sealift Ships (FSS), Maritime Prepositioning Ships (MPS), Afloat Prepositioning Ships (APS), and the Ready Reserve Force (RRF) (Willie 1992). A summary of the general characteristics of these ships, obtained from Hura (1993) and Prezelin (1990), is shown in Table 3 below.

Table 3. General Ship Characteristics (Government-Controlled Ships)

Source		Speed (kts)	Quantity (Available)	Capacity (sq ft)	Activation (days)	Load/Unload (days)
FSS		30	8	165,000	4	2
MPS		19	13	125,000	1	2
APS		16-22	8	82,000	1	2
RRF:	Ro/Ro	20	17	110,000	5	2
	B/B	17	51	70,000	5/10/20	4

There are other ships in the U.S. Government-controlled ship fleet not shown above but they are more specialized (hospital ships, barges, tankers, etc.) and therefore have little to do with the transport of heavy Army unit equipment. The 8 FSS ships are the

most suitable for transport of unit equipment and together can move the equivalent of about one heavy Army division. They are the fastest ships in the fleet with the largest capacity. They are also configured as Roll-on/Roll-off (Ro/Ro) ships which make the loading and unloading of unit vehicles, such as tanks, much easier. They are primarily located on the east coast of the U.S. with an activation time of four days. The MPS ships are the most responsive ships in the fleet. They are pre-loaded with unit equipment for the Marine Corps and have the ability to set sail in one day. The 13 MPS ships are broken into three squadrons: MPS Squadron 1 (4 ships) is located on the U.S. east coast; MPS Squadron 2 (5 ships) is located at Diego Garcia in the Indian Ocean; and MPS Squadron 3 (4 ships) is located in Guam. Each MPS squadron carries the equipment and supplies necessary to sustain a Marine expeditionary brigade (MEB) for 30 days. During Desert Shield, MPS Squadrons 2 and 3 were alerted and sent to Saudi Arabia. After the delivery of the Marine equipment, these ships are available to deliver Army equipment. The APS ships are also pre-loaded and forward-positioned. These ships carry ammunition and supplies for the Army and the Air Force. They are primarily located at Diego Garcia along with MPS Squadron 2. The APS ships turn out to be less useful for transport of Army equipment and are used primarily to deliver ammunition and other supplies. The last source of ships, the RRF, consists of a wide variety of ship types in various stages of readiness. The 17 Ro/Ro ships in the RRF were immediately alerted during Desert Shield due to their facility with loading and unloading of unit equipment. They are berthed throughout the U.S. Of the available 51 breakbulk (B/B) ships of the RRF, only about 27

were used during the surge phase of Desert Shield due to maintenance and manning problems (U.S. Congress 1991).

The information provided in U.S. Congress (1991) identify by name the ships and units alerted during the first 90 days of Operation Desert Shield. This information was used to develop a realistic data set. Data Set I incorporates the 61 ships used (less the commercial U.S flag and foreign flag ships) during Desert Shield's surge phase. The characteristics of these ships are summarized in Table 4 below. Data for these ships and their initial locations during Desert Shield was obtained from U.S. Congress (1991). Data concerning specific ship characteristics was obtained from Prezelin (1990). A comprehensive listing of the ships is contained in Appendix B.

Table 4. Available Ships

Source		Capacity (sq ft)	Quantity (Available)
FSS		165,000	8
MPS		125,000	9
RRF:	Ro/Ro	110,000	17
	B/B	70,000	27
Total			61

To make the problem manageable, crew and maintenance issues were ignored and the capacity of each of the ships was assumed to be the same. A common capacity for all ships was obtained by taking a weighted average of available ship capacities. This average was about 100,000 square feet. Units were then broken into increments (or jobs) of 100,000 square feet based upon their cargo requirements. Thus each unit increment

corresponds to one shipload. The number of shiploads per unit is summarized in Table 5 below. Table 5 also contains unit types and due dates.

Table 5. Additional Unit Characteristics

Unit	Cargo Requirement (sq ft)	Shiploads (x100,000 sq ft)	Unit Types	Unit Due Dates (Days)
24th MECH	2,375,000	24	1	30
101st AASLT	406,250	4	2	40
3rd ACR	770,000	8	3	50
1st CAV	2,415,000	24	4	60
COSCOM	4,033,750	8,6,4,10,12	1,2,3,5,6	30,40,50,80,90
Total	10,000,000	100		

Note: COSCOM figures are broken down by types.

Recall that unit types allow for further refinement of precedence within a unit. This allows for finer control of unit flow over and above the macro level unit priority. The major unit priorities will be considered a hard constraint. However, within a particular unit, precedence will be considered a soft constraint and will be modeled using unit types. Unit due dates represent the desired completion dates for each type of unit. These were developed based upon Desert Shield experience (U.S. Congress 1991). The deviation of unit completion times from these dates will be used as a benchmark for determining how good a schedule is developed.

The final element to be considered in defining the problem concerns the possible routes a ship may take. During the initial phase in Desert Shield, the route taken from the U.S. was almost exclusively through the Suez Canal (U.S. Congress 1991). The exception to this was the 3rd MPS Squadron in Guam which took the direct route to the

west to reach Saudi Arabia on its first trip. For the purposes of this problem the unconstrained availability of the Suez Canal will be assumed and this route will be the route used by all ships departing from U.S. ports.

The key elements of the problem have been described above. The problem can now be specifically described in terms of the facts and assumptions listed above. The problem is to determine the most effective manner in which the given shipping assets can be employed so as to transport the required units, according to their priority, from their seaport of embarkation (SPOE) to Ad Dammam port. The total deviation of the units from their desired due dates will be used as a measure of solution effectiveness.

Problem Formulation

The strategic sealift scheduling problem can be readily expressed in the language of the job shop. The unit increments corresponding to the capacity of a single ship can be represented as individual jobs to be processed by the ships. At the outset, the jobs are arranged in precedence or priority order based upon readiness and operational requirements. The ships can be represented as machines with different processing rates due to the different cruising speeds of ship types as well as the different speeds of an individual ship when empty or full. Thus the available ships can be viewed as multiple parallel machines with machine-dependent processing times. There is an added difficulty due to the ballast runs (or empty return trips) ships make between jobs. This deadheading can be represented by a changeover or setup cost which is assessed when changing from one job to another. Additionally, all the ships are not ready at the start of the problem

due to their readiness posture. The idea is to determine the best or near best match of machines to jobs (ships to units) so as to minimize the sum of job (unit) tardiness.

In short, this problem can be described as scheduling a set of jobs with precedence constraints on parallel unrelated machines with sequence dependent changeovers to minimize the sum of job tardiness. Additional weight should be given to higher priority units which do not meet their due date. Therefore a weighted tardiness objective will be used with heavier weighting going to tardy jobs with higher priority. Thus the objective is to minimize the sum of the weighted job tardiness. The following is an application of Guinet's (1991) model for scheduling a textile production system to the strategic sealift scheduling problem.

Let N be the number of jobs to be processed with indices i and j. Let M be the number of machines available with index k. Consider the time required to process a job on a particular machine. The total processing time for job j on machine k consists of the time required to setup the machine for the job plus the time to process the job on the machine. The setup time depends upon the position of job j on machine k and the characteristics of job j. Let $s(i,j)$ represent the setup or changeover time to process job j after job i on machine k. If job j is the first job assigned to machine k (i.e., $i=0$) the time, $s(0,j)$, is the time required to "warm-up" the machine plus the time to setup the machine for job j. This corresponds to the time required to ready a ship for sea (number of activation days) plus the time required for the ship to travel to its first port destination. For all subsequent jobs on machine k (i.e., job j is processed after some job i on machine k), the time, $s(i,j)$, reflects a changeover time incurred in preparing machine k for job j.

This corresponds to the deadheading of ships from the destination port to the next unit port. Since all deadheading trips originate from the same location (Ad Dammam port), the setup cost depends only on the unit's initial location (SPOE) for a given ship.

Let $p(i,k)$ represent the time required to process job i on machine k. The processing time consists of the time to load the job on the machine, the time to actually process the job on the machine, and the time to unload the job from the machine. Loading and unloading times are machine dependent. This means that the loading and unloading times for all jobs on machine k will be the same and therefore can be incorporated into the processing time for the job. Actual processing times for a job on machine k depend upon the particular job being processed. This is because for a given ship, the time required to transport a unit depends upon the distance of the unit's location (SPOE) to the unit's destination (SPOD). For simplicity, the processing time, $p(i,k)$, will consist of the loading/unloading times and the processing time together.

The total time required to process job j on machine k can now be expressed as:

$$p_{tot}(j,k) = s(i,j) + p(j,k), \quad \text{for } i \neq j;\ i=0,\dots,N;\ j=1,\dots,N;\ k=1,\dots,M.$$

The completion time of job j is a function of the machine which processed it and the position of the job on that machine. Let $c(i)$ represent the completion time of job i. If job j is serviced after job i on the same machine, the completion time of job j is:

$$\begin{aligned} c(j) &= c(i) + s(i,j) + p(j,k) \\ &= c(i) + p_{tot}(j,k), \quad \text{for } i \neq j;\ i=0,\dots,N;\ j=1,\dots,N;\ k=1,\dots,M. \end{aligned}$$

The introduction of a binary variable is necessary to extend these relationships to multiple non-identical machines. Let $x(i,j,k)$ be 1 if job j is processed directly after job i on machine k and 0 otherwise. Therefore $x(0,j,k)=1$ means that job j is the first job processed on machine k and $x(j,0,k)=1$ means that job j is the last job processed on machine k.

Finally, let $z(i)$ represent the tardiness of job i, $d(i)$ represent the due date of job i, and $w(i)$ represent the weight attributed to job i. The job tardiness, $z(i)$, can be expressed as:

$$z(i) = Max\{0, c(i) - d(i)\},\ i=1,\dots, N.$$

The overall objective of minimizing the weighted sum of job tardiness is then expressed as:

$$Min\ Z = \sum_{i=1}^{N} w(i)\, z(i)$$

Using the model proposed by Guinet (1991), the problem can now be formulated as follows:

Model

$$\text{Minimize} \sum_{i=1}^{N} w(i)\, z(i) \tag{1}$$

subject to:

$$\sum_{\substack{i=0 \\ i\neq j}}^{N} \sum_{k=1}^{M} x(i,j,k) = 1 \qquad \forall j = 1,\dots,N, \tag{2}$$

$$\sum_{\substack{i=0 \\ i\neq h}}^{N} x(i,h,k) - \sum_{\substack{j=0 \\ j\neq h}}^{N} x(h,j,k) = 0 \qquad \forall h = 1,\dots,N,\ \forall k = 1,\dots,M, \tag{3}$$

$$c(j) \geq c(i) + \sum_{k=1}^{M} x(i,j,k) * (s(i,j) + p(j,k)) + \left[\sum_{k=1}^{M} x(i,j,k) - 1 \right] * HV, \tag{4}$$

$$
\begin{aligned}
&\qquad\qquad\qquad\qquad \forall i = 0,\ldots,N,\ \forall j = 1,\ldots,N, \\
&c(i) - d(i) \le z(i) \quad \forall i = 1,\ldots,N \qquad\qquad\qquad\qquad\qquad\qquad (5) \\
&x(i,j,k) \in \{0,1\} \quad \forall i = 1,\ldots,N,\ \forall j = 1,\ldots,N,\ \forall k = 1,\ldots,M \\
&z(i) \ge 0,\ c(i) \ge 0, \quad \forall i = 1,\ldots,N;\ c(0) = 0,
\end{aligned}
$$

where

N=number of jobs;
M=number of machines;
$p(i,k)$=processing time to complete job i with machine k;
$s(i,j)$=setup time required to process job j after job i on the same machine;
$s(0,j)$=setup time required to process job j first on a machine;
the indices h,i,j correspond to jobs, and k to machines;
the index 0 corresponds to the problem state prior to scheduling;
$x(i,j,k)$=1 if job j is processed directly after job i on machine k, and 0 otherwise;
$x(0,j,k)$=1 if job j is the first job to be processed on machine k, and 0 otherwise;
$x(j,0,k)$=1 if job j is the last job to be processed on machine k, and 0 otherwise;
$c(i)$=completion date of job i;
$d(i)$=due date of job i;
$z(i)$= tardiness of job i;
HV=a scalar chosen to be larger than the maximum completion date of any feasible solution.

The objective (1) minimizes the weighted sum of job tardiness. Constraints (2) ensure that a job is processed only once. Constraints (3) ensure that each job has a predecessor and a successor. Constraints (4) permit the calculation of the job completion times and prevent a job from being the predecessor and successor of the same job. Constraints (5) allow the calculation of job tardiness.

Problem Complexity

The complexity of this problem has a direct impact upon the development of any algorithm. There are two issues concerning the complexity of the strategic sealift scheduling problem. First, Guinet (1991) indicates that the job-dependent changeover problem with weighted tardiness defines an NP-complete problem. NP-complete

problems are very difficult to solve for realistic problems like the one studied here. NP is the class of problems for which it is possible to guess a feasible schedule and then check to see if this schedule is feasible in polynomial time (French 1982). French defines an NP-complete problems as a subclass of NP problems which cannot be solved in polynomial time. In a related way, the number of possible schedules for this problem is immense. As an example, consider the problem of scheduling just 3 machines to handle 8 jobs. All told, there are 6,561 possible schedules to be considered. For a realistic problem such as the one described here, the number of possible schedules is 61^{100} or $3x10^{178}$! In general there are M^N possible schedules. Clearly an exhaustive search of the possible schedules is not feasible for a problem of this size because the time required to solve the problem is impractical. This is known as the combinatorial explosion problem (Rich and Knight 1991). These two issues imply that some heuristic must be developed to solve this problem in a reasonable amount of time. The intent of the heuristic is to obtain a "good" solution to the problem without searching the entire solution space. The next section outlines the development of the heuristic used to obtain a solution to the strategic sealift scheduling problem.

Algorithm Development

There are many potential heuristics which can be applied to this type of problem. The method chosen combines a sequential assignment procedure for identical parallel machines adapted from Dogramaci and Surkis (1979) with a schedule generation technique developed by Guinet (1991) for non-identical parallel textile machines.

The sequential assignment procedure of Dogramaci and Surkis (1979) consists of the following steps:

(1) Order the jobs according to a priority rule.

(2) For the jobs not yet assigned to a machine, choose the first one in the priority list, assign it to the machine which yields the earliest completion date for the job, and remove the job from the list. Repeat this step until all the jobs are assigned.

(3) For each machine re-sequence the assigned jobs on that machine so as to minimize the sum of weighted job tardiness.

The schedule generation technique of Guinet (1991) creates new schedules by making pairwise exchanges of the jobs on a given schedule. This technique seeks to improve the solution by generating and testing each of these new schedules against the objective of minimizing the sum of weighted job tardiness.

The modified heuristic, called SAA by Guinet (1991), consists of a combination of these two techniques. This new heuristic consists of the following steps:

(1) Arrange the jobs in priority order.

(2) Assign the jobs to the machines according to this priority by selecting the machine which minimizes a job's completion date.

(3) Improve the solution by pairwise exchange of the jobs within and among machines. Exchanges are allowed only if they result in a smaller weighted tardiness objective.

(4) Continue making pairwise exchanges until no improvement can be made.

The sequential assignment portion of the SAA algorithm has intuitive appeal. Given a set of ordered jobs it attempts to assign those jobs one at a time so that each job is assigned a machine which minimizes that job's completion date. This should provide a good starting solution to the problem. The pairwise exchange of jobs seeks to improve upon this initial solution by making pairwise exchanges of jobs until no further improvement can be made. Exchanges are made until no improvement can be found.

Algorithm Implementation and Testing

The SAA heuristic was implemented using an algorithm prototyping language called MOR/ML developed by Deuermeyer, Curry, and Feldman (1991). All programs were run on a Dell 486DX2/66Mhz microcomputer. The MOR/ML code for the SAA heuristic is attached at Appendix F. Data Set I, the problem data set developed using the Desert Shield experience as a guideline, is contained in Appendices A and B. Data Set I consists of 61 machines (ships) and 100 jobs (units). This is the real world problem developed using the experience of Desert Shield as a guide. Additional smaller data sets were developed to measure algorithm performance. These data sets, called Data Sets II and III, are contained in Appendices D and E respectively. The problem size for these data sets are 3 machine and 8 jobs.

The SAA heuristic was executed for Data Sets I, II, and, III. To test the results of the SAA heuristic for Data Set I, a set of 1000 schedules were randomly generated and evaluated against the weighted tardiness objective. For the smaller problem sizes (Data Sets II and III) exhaustive enumeration was feasible and therefore was used to obtain the

optimal schedules subject to the weighted tardiness objective. The program used for obtaining the optimal solution using exhaustive enumeration is contained in Appendix G. The results obtained using the SAA heuristic for each of these problems sets are presented in the next section.

RESULTS

The results for Data Sets I, II, and III are shown in Table 6 below. The schedule obtained for Data Set I using the SAA heuristic is contained in Appendix C. This schedule resulted in a weighted tardiness objective of 663. The best weighted tardiness objective obtained in 1000 randomly generated schedules was 2911. Thus the SAA heuristic resulted in a weighted tardiness objective over 4 times better than the best randomly generated schedule. The results for Data Sets II and III are also encouraging. The SAA heuristic achieved the optimal result for Data Set II and Data Set III. The optimal schedules are contained in Appendices D and E.

Table 6. Weighted Tardiness Objective Results

Data Set	SAA Heuristic	Exhaustive Search	Random Schedules
I	663	N/A	2,911
II	262	262	N/A
III	373	373	N/A

The results indicate that the SAA heuristic achieves a reasonably good solution to the weighted tardiness objective. The results also show that the heuristic can find the optimal result for some problems. Perhaps the greatest advantage of the SAA heuristic is that it obtains a good solution rapidly. Table 7 below shows the amount of time required to reach a solution for the various techniques and problem sizes.

Table 7. Program Run Times (Minutes)

Data Set	Machines	Jobs	SAA Heuristic	Exhaustive Search
I	61	100	104.08	N/A
II	3	8	0.16	14.12
III	3	8	0.16	14.12

Although the SAA heuristic takes over 104 minutes to run for Data Set I, the alternative using an exhaustive search is completely infeasible.

CONCLUSIONS

A detailed analysis was conducted of the strategic sealift problem as it pertains to the U.S. Army's initial surge requirement. A heuristic was developed and implemented on a realistic problem. The SAA heuristic performed very well against randomly generated schedules and, for smaller problems, actually achieved the optimal solution.

The SAA heuristic permits within unit priorities to be used in assessing the desirability of a schedule thus providing greater flexibility in managing the flow of units to the theater of operations. A key use of the heuristic pertains to the issue of deciding upon when and how many ships should be contracted by the government to make up for shortfalls in the strategic sealift capability. Based upon the unit due dates and the weighted tardiness objective, additional ships can be added and their impact upon unit tardiness can be assessed to determine the additional ships required to meet the desired unit closure dates. The contracting decision turned out to be critical one during Desert Shield due to breakdowns in several ships and the addition of units to the deployment list (U.S. Congress 1991).

The military planning cells responsible for strategic sealift employ a variety of models and tools to analyze the strategic sealift problem (Schank et al. 1991). A number of these tools employ large computerized models which are often unwieldy and somewhat restricted in scope. This research indicates that modeling the strategic sealift problem can be extremely complex. This research focused on a very narrow aspect of the problem. The study of this problem indicates that mathematical programming alone will not solve problems of this nature. A recommended strategy for developing better models is to

develop a model using a hybrid of mathematical programming and knowledge-based modeling techniques. Concurrent use of these two techniques would perhaps enable better modeling of the uncertainty inherent in scheduling sealift and allow for broader objective functions than the narrow one considered here.

REFERENCES

Bellmore, M., G. Bennington, and S. Lubore. 1971. A Multivehicle Tanker Scheduling Problem. *Transportation Science* **5**, 36-47.

Defense Mapping Agency. 1985. Distances Between Ports. Pub. 151. GPO, Washington, D.C.

Deuermeyer, B.L., G.L. Curry, and R.M. Feldman. 1991. User's Guide for MOR/ML: An Algorithm Prototyping Language for Operations Research. Department of Industrial Engineering, Texas A&M University, College Station, Texas.

Dogramaci, A., and J. Surkis. 1979. Evaluation of a Heuristic for Scheduling Independent Jobs on Parallel Identical Processors. *Management Science*, **25**, 1208-1216.

Fisher, M.L., and M.B. Rosenwein. 1989. An Interactive Optimization System for Bulk-cargo Ship Scheduling. *Naval Research Logistics Quarterly* **36**, 27-42.

French, S. 1982. *Sequencing and Scheduling: An Introduction to the Mathematics of the Job Shop*. Ellis Horwood, Chichester, England

Guinet, A. 1991. Textile Production Systems: A Succession of Non-identical Parallel Processor Shops. *Journal of the Operational Research Society*, **42**, 655-671.

Hura, M., J. Matsumura, and R. Robinson. 1993. An Assessment of Alternative Transports for Future Mobility Planning. Rand, Santa Monica, California.

Laderman, J., L. Glieberman, and J. F Egan. 1966. Vessel Allocation by Linear Programming. *Naval Research Logistics Quarterly* 315-320.

McKay, M.D., and H.O. Hartley. 1974. Computerized Scheduling of Seagoing Tankers. *Naval Research Logistics Quarterly* **21**, 255-264.

Prezelin, B. (ed.). 1991. *Combat Fleets of the World 1990/1991: Their Ships, Aircraft, and Armament*. Naval Institute Press, Annapolis, Maryland.

Rich, E., and K. Knight. 1991. *Artificial Intelligence*, 2nd ed. McGraw-Hill, NewYork.

Ronen, D. 1983. Cargo Ships Routing and Scheduling: Survey of Models and Problems. *European Journal of Operational Research* **12**, 119-126.

Schank, J., M. Mattock, G. Sumner et al. 1991. A Review of Strategic Mobility Models and Analysis. Rand, Santa Monica, California.

U.S. Army, Military Traffic Management Command Transportation Engineering Agency. 1991. Deployment Planning Guide. GPO, Washington, D.C.

U.S. Congress, House Committee on Merchant Marine and Fisheries. 1990. Our Nation's Capability to Meet Sealift Requirements Caused by American Deployment to the Persian Gulf. 101st Cong., 2nd Sess., 18 and 26 September.

U.S. Congress, House Committee on Merchant Marine and Fisheries. 1991. The Performance of our Nation's Sealift Capabilities During Operation Desert Shield/Desert Storm and Future Promotional Policies to Improve the Commercial Merchant Marine Fleet. 102nd Cong., 1st Sess., 23 April and 21 May.

Willie, J.F. 1992. The U.S. Strategic Mobility Posture- A Critical Factor to Support National Security Objectives. U.S. Army War College Military Studies Program Paper, U.S. Army War College, Carlisle Barracks, Pennsylvania.

APPENDICES

APPENDIX A

UNIT DATA

UNIT NO	UNIT NAME	SPOE	PRIORITY NUMBER	TYPE	DUE DATE (DAYS)
1	101AA1	MOBL	33	2	40
2	101AA2	MOBL	34	2	40
3	101AA3	MOBL	35	2	40
4	101AA4	MOBL	36	2	40
5	24MECH1	SVNH	1	1	30
6	24MECH2	SVNH	2	1	30
7	24MECH3	SVNH	3	1	30
8	24MECH4	SVNH	4	1	30
9	24MECH5	SVNH	5	1	30
10	24MECH6	SVNH	6	1	30
11	24MECH7	SVNH	7	1	30
12	24MECH8	SVNH	8	1	30
13	24MECH9	SVNH	9	1	30
14	24MECH10	SVNH	10	1	30
15	24MECH11	SVNH	11	1	30
16	24MECH12	SVNH	12	1	30
17	24MECH13	SVNH	13	1	30
18	24MECH14	SVNH	14	1	30
19	24MECH15	SVNH	15	1	30
20	24MECH16	SVNH	16	1	30
21	24MECH17	SVNH	17	1	30
22	24MECH18	SVNH	18	1	30
23	24MECH19	SVNH	19	1	30
24	24MECH20	SVNH	20	1	30
25	24MECH21	SVNH	21	1	30
26	24MECH22	SVNH	22	1	30
27	24MECH23	SVNH	23	1	30
28	24MECH24	SVNH	24	1	30
29	1CD1	BMNT	55	4	60
30	1CD2	BMNT	56	4	60

APPENDIX A

UNIT DATA

UNIT NO	UNIT NAME	SPOE	PRIORITY NUMBER	TYPE	DUE DATE (DAYS)
31	1CD3	BMNT	57	4	60
32	1CD4	BMNT	58	4	60
33	1CD5	BMNT	59	4	60
34	1CD6	BMNT	60	4	60
35	1CD7	BMNT	61	4	60
36	1CD8	BMNT	62	4	60
37	1CD9	BMNT	63	4	60
38	1CD10	BMNT	64	4	60
39	1CD11	BMNT	65	4	60
40	1CD12	BMNT	66	4	60
41	1CD13	BMNT	67	4	60
42	1CD14	BMNT	68	4	60
43	1CD15	BMNT	69	4	60
44	1CD16	BMNT	70	4	60
45	1CD17	BMNT	71	4	60
46	1CD18	BMNT	72	4	60
47	1CD19	BMNT	73	4	60
48	1CD20	BMNT	74	4	60
49	1CD21	BMNT	75	4	60
50	1CD22	BMNT	76	4	60
51	1CD23	BMNT	77	4	60
52	1CD24	BMNT	78	4	60
53	3ACR1	BMNT	43	3	50
54	3ACR2	BMNT	44	3	50
55	3ACR3	BMNT	45	3	50
56	3ACR4	BMNT	46	3	50
57	3ACR5	BMNT	47	3	50
58	3ACR6	BMNT	48	3	50
59	3ACR7	BMNT	49	3	50
60	3ACR8	BMNT	50	3	50

APPENDIX A

UNIT DATA

UNIT NO	UNIT NAME	SPOE	PRIORITY NUMBER	TYPE	DUE DATE (DAYS)
61	COSCOM1	WILM	25	1	30
62	COSCOM2	WILM	26	1	30
63	COSCOM3	WILM	27	1	30
64	COSCOM4	WILM	28	1	30
65	COSCOM5	WILM	29	1	30
66	COSCOM6	WILM	30	1	30
67	COSCOM7	WILM	31	1	30
68	COSCOM8	WILM	32	1	30
69	COSCOM9	WILM	37	2	40
70	COSCOM10	WILM	38	2	40
71	COSCOM11	WILM	39	2	40
72	COSCOM12	WILM	40	2	40
73	COSCOM13	WILM	41	2	40
74	COSCOM14	WILM	42	2	40
75	COSCOM15	WILM	51	3	50
76	COSCOM16	WILM	52	3	50
77	COSCOM17	WILM	53	3	50
78	COSCOM18	WILM	54	3	50
79	COSCOM19	BMNT	79	5	80
80	COSCOM20	BMNT	80	5	80
81	COSCOM21	BMNT	81	5	80
82	COSCOM22	BMNT	82	5	80
83	COSCOM23	BMNT	83	5	80
84	COSCOM24	BMNT	84	5	80
85	COSCOM25	BMNT	85	5	80
86	COSCOM26	BMNT	86	5	80
87	COSCOM27	BMNT	87	5	80
88	COSCOM28	BMNT	88	5	80
89	COSCOM29	BMNT	89	6	90
90	COSCOM30	BMNT	90	6	90

APPENDIX A

UNIT DATA

UNIT NO	UNIT NAME	SPOE	PRIORITY NUMBER	TYPE	DUE DATE (DAYS)
91	COSCOM31	BMNT	91	6	90
92	COSCOM32	BMNT	92	6	90
93	COSCOM33	BMNT	93	6	90
94	COSCOM34	BMNT	94	6	90
95	COSCOM35	BMNT	95	6	90
96	COSCOM36	BMNT	96	6	90
97	COSCOM37	BMNT	97	6	90
98	COSCOM38	BMNT	98	6	90
99	COSCOM39	BMNT	99	6	90
100	COSCOM40	BMNT	100	6	90

APPENDIX B

SHIP DATA

SHIP_NO	NAME	INITIAL PORT	ACTIVATION (DAYS)	SPEED(FULL) (KNOTS)	SPEED(EMPTY) (KNOTS)	SOURCE	LOADING (DAYS)
1	ALGOL	SVNH	4	30.1	33	FSS	2
2	ALTAIR	SVNH	4	30.1	33	FSS	2
3	ANTARES	SVNH	4	30.1	33	FSS	2
4	BELLATRIX	SVNH	4	30.1	33	FSS	2
5	CAPELLA	SVNH	4	30.1	33	FSS	2
6	DENEBOLA	SVNH	4	30.1	33	FSS	2
7	POLLUX	SVNH	4	30.1	33	FSS	2
8	REGULUS	SVNH	4	30.1	33	FSS	2
9	BONNEYMAN	DAMM	8	17.2	18.5	MPS	2
10	HAUGE	DAMM	8	17.2	18.5	MPS	2
11	BAUGH	DAMM	8	17.2	18.5	MPS	2
12	ANDERSON	DAMM	8	17.2	18.5	MPS	2
13	FISHER	DAMM	8	17.2	18.5	MPS	2
14	LUMMUS	DAMM	16	17.7	18.8	MPS	2
15	WILLIAMS	DAMM	16	17.7	18.8	MPS	2
16	LOPEZ	DAMM	16	17.7	18.8	MPS	2
17	BUTTON	DAMM	16	17.7	18.8	MPS	2
18	COMET	PTLD	5	18	19	RRF	2
19	EDMONT	PTLD	5	19	20	RRF	2
20	LAMBERT	JRRF	5	19	20	RRF	2
21	LOBOS	JRRF	5	19	20	RRF	2
22	HENRY	JRRF	5	21	22	RRF	2
23	HUDSON	JRRF	5	21	22	RRF	2
24	HORN	SFRN	5	21	22	RRF	2
25	DECISION	JRRF	5	22	23	RRF	2
26	DIAMOND	JRRF	5	22	23	RRF	2
27	DOMINGO	JRRF	5	22	23	RRF	2
28	DUCATO	SPDO	5	22	23	RRF	2
29	METEOR	SPDO	5	22	23	RRF	2
30	DOUGLAS	JACK	5	22	23	RRF	2
31	INSCRIPTION	MOBL	5	23	24	RRF	2
32	ISABEL	PTLD	5	23	24	RRF	2
33	JUPITER	PTLD	5	23	24	RRF	2

APPENDIX B

SHIP DATA

SHIP_NO	NAME	INITIAL PORT	ACTIVATION (DAYS)	SPEED(FULL) (KNOTS)	SPEED(EMPTY) (KNOTS)	SOURCE	LOADING (DAYS)
34	CALLAGHAN	JRRF	20	26	27	RRF	2
35	CAPE NOME	JRRF	5	23.6	24.6	RRF	4
36	CAPE GIBSON	SFRN	5	21	22	RRF	4
37	CAPE GIRARDEAU	SFRN	5	21	22	RRF	4
38	CAPE JOHNSON	JRRF	5	20	21	RRF	4
39	CAPE JUBY	JRRF	5	20	21	RRF	4
40	DEL MONTE	BMNT	5	18.6	19.6	RRF	4
41	DEL VALLE	BMNT	10	18.6	19.6	RRF	4
42	DEL VIENTO	BMNT	5	18.6	19.6	RRF	4
43	GULF BANKER	BMNT	10	18	19	RRF	4
44	GULF FARMER	BMNT	10	18	19	RRF	4
45	GULF MERCHANT	BMNT	10	18	19	RRF	4
46	GULF SHIPPER	BMNT	5	18	19	RRF	4
47	GULF TRADER	BMNT	5	18	19	RRF	4
48	CAPE ALAVA	JRRF	5	20	21	RRF	4
49	CAPE ALEXANDER	JRRF	5	20	21	RRF	4
50	CAPE CHALMERS	BMNT	10	18	19	RRF	4
51	CAPE CHARLES	BMNT	10	18	19	RRF	4
52	CAPE CLEAR	BMNT	10	18	19	RRF	4
53	CAPE COD	BMNT	10	18	19	RRF	4
54	PIONEER COMMANDER	BMNT	10	21	22	RRF	4
55	PIONEER CONTRACTOR	BMNT	10	21	22	RRF	4
56	PIONEER CRUSADER	BMNT	10	21	22	RRF	4
57	SANTA ANA	BMNT	10	20	21	RRF	4
58	BANNER	JRRF	10	18.5	19.5	RRF	4
59	COURIER	JRRF	10	18.5	19.5	RRF	4
60	CAPE CATAWBA	BMNT	10	19	20	RRF	4
61	WASHINGTON	BMNT	10	16.5	17.5	RRF	4

APPENDIX C

SCHEDULE GENERATED USING SAA HEURISTIC (DATA SET I)

SHIP_NO	SHIP NAME	UNIT(S) DELIVERED		
		FIRST	SECOND	THIRD
1	ALGOL	24MECH1	COSCOM17	COSCOM25
2	ALTAIR	24MECH2	COSCOM18	COSCOM26
3	ANTARES	24MECH3	1CD1	COSCOM32
4	BELLATRIX	24MECH4	1CD2	COSCOM33
5	CAPELLA	24MECH5	1CD3	COSCOM34
6	DENEBOLA	24MECH6	1CD4	COSCOM35
7	POLLUX	24MECH7	1CD5	COSCOM36
8	REGULUS	24MECH8	1CD6	COSCOM37
9	BONNEYMAN	1CD7		
10	HAUGE	1CD8		
11	BAUGH	1CD9		
12	ANDERSON	1CD10		
13	FISHER	1CD11		
14	LUMMUS	1CD12		
15	WILLIAMS	1CD13		
16	LOPEZ	1CD14		
17	BUTTON	1CD15		
18	COMET	COSCOM15		
19	EDMONT	COSCOM11		
20	LAMBERT	24MECH16	COSCOM23	
21	LOBOS	24MECH17	COSCOM24	
22	HENRY	24MECH19	1CD21	
23	HUDSON	24MECH14	1CD22	
24	HORN	24MECH13	COSCOM27	
25	DECISION	24MECH10	1CD18	
26	DIAMOND	24MECH11	1CD19	
27	DOMINGO	24MECH12	1CD20	
28	DUCATO	24MECH23	COSCOM21	
29	METEOR	24MECH24	COSCOM22	
30	DOUGLAS	24MECH9	1CD16	

APPENDIX C

SCHEDULE GENERATED USING SAA HEURISTIC (DATA SET I)

SHIP_NO	SHIP NAME	UNIT(S) DELIVERED		
		FIRST	SECOND	THIRD
31	INSCRIPTION	24MECH15	1CD17	
32	ISABEL	24MECH21	COSCOM19	
33	JUPITER	24MECH22	COSCOM20	
34	CALLAGHAN	COSCOM8	1CD24	
35	CAPE NOME	24MECH18	1CD23	
36	CAPE GIBSON	COSCOM12		
37	CAPE GIRARDEAU	COSCOM13		
38	CAPE JOHNSON	COSCOM5	COSCOM28	
39	CAPE JUBY	COSCOM1	COSCOM29	
40	DEL MONTE	COSCOM3		
41	DEL VALLE	COSCOM14		
42	DEL VIENTO	COSCOM4		
43	GULF BANKER	3ACR2		
44	GULF FARMER	3ACR3		
45	GULF MERCHANT	3ACR4		
46	GULF SHIPPER	24MECH20		
47	GULF TRADER	COSCOM7		
48	CAPE ALAVA	COSCOM6	COSCOM30	
49	CAPE ALEXANDER	COSCOM2	COSCOM31	
50	CAPE CHALMERS	3ACR5		
51	CAPE CHARLES	3ACR6		
52	CAPE CLEAR	3ACR7		
53	CAPE COD	3ACR8		
54	PIONEER COMMANDER	101AA1	COSCOM38	
55	PIONEER CONTRACTOR	101AA2	COSCOM39	
56	PIONEER CRUSADER	101AA3	COSCOM40	
57	SANTA ANA	101AA4		
58	BANNER	COSCOM9		
59	COURIER	COSCOM10		
60	CAPE CATAWBA	3ACR1		
61	WASHINGTON	COSCOM16		

APPENDIX D

SCHEDULE GENERATED USING SAA HEURISTIC (DATA SET II)

SHIP_NO	SHIP NAME	UNIT(S) DELIVERED			
		FIRST	SECOND	THIRD	FOURTH
1	ALGOL	24MECH1	101AA2	101AA1	3ACR2
2	LAMBERT	24MECH2	COSCOM9		
3	PIONEER CRUSADER	COSCOM10	3ACR1		

MAKESPAN=109

WEIGHTED TARDINESS OBJ=262

APPENDIX E

SCHEDULE GENERATED USING SAA HEURISTIC (DATA SET III)

SHIP_NO	SHIP NAME	UNIT(S) DELIVERED			
		FIRST	SECOND	THIRD	FOURTH
1	ALTAIR	24MECH1	COSCOM14	3ACR7	COSCOM19
2	COMET	COSCOM1	COSCOM17		
3	CAPE COD	101AA4	1CD1		

MAKESPAN=108

WEIGHTED TARDINESS OBJ=373

APPENDIX F

SAA ALGORITHM PROGRAM

```
RealFormat[10,4];
out1=Open["saa7pt3.out","w"];
out2=Open["saa7pt4.out","w"];
NumSup = 4;    [*NUMBER OF SPOES*]
NumDest = 1;   [*NUMBER OF SPODS*]
NumJobs = 100;  [*NUMBER OF UNITS*]
NumMachs = 61;  [*NUMBER OF SHIPS*]
JT=ShipSeq=Table[0,{i,NumJobs}];
LJ = Table[0,{i,NumMachs}];  [*Initialize Latest Job to Index 0*]
JN = Table[0,{i,NumMachs}];  [*Initialize Machine Job Number to 0*]
[* Time to SPOEs by ship type for first trip*]
FirstTrip={{0,0,0,0,0,0,0,0,19,19,19,19,19,19,19,19,19,12,11,1,1,0,0,10,0,0,0,9,9,0,2,9,9,0,0,10,1
0,0,0,3,3,3,3,3,3,3,3,0,0,3,3,3,3,3,3,3,3,1,1,3,3},{0,0,0,0,0,0,0,0,21,21,21,21,21,20,20,20,20,12,11
,1,1,1,1,9,1,1,1,8,8,0,2,9,9,1,1,9,9,1,1,3,3,3,3,3,3,3,3,1,1,3,3,3,3,2,2,2,3,1,1,3,3},{1,1,1,1,1,1,1,2
2,22,22,22,22,21,21,21,21,11,11,3,3,3,3,9,3,3,3,8,8,2,0,9,9,2,2,9,9,3,3,0,0,0,1,1,1,1,1,3,3,1,1,1,1,0
,0,0,0,3,3,0,1},{2,2,2,2,2,2,2,2,22,22,22,22,22,22,22,22,22,11,11,4,4,3,3,9,3,3,3,8,8,2,0,9,9,3,3,9,
9,3,3,0,0,0,0,0,0,0,0,3,3,0,0,0,0,0,0,0,0,4,4,0,0}};

[*Processing time for a job at location i on machine k*]
TransT={{16,16,16,16,16,16,16,16,25,25,25,25,25,24,24,24,24,24,23,23,23,21,21,21,20,20,20,20
,20,20,20,20,20,18,23,25,25,26,26,27,27,27,28,28,28,28,28,26,26,28,28,28,28,25,25,25,26,27,27,
27,30},{17,17,17,17,17,17,17,17,26,26,26,26,  26,26,26,26,26,25,24,24,24,22,22, 22,
21,21,21,21,21,21,21,21,21,19,24,26,26,   27,   27,   29,   29,   29,   29,29,   29,   29,
29,   27,   27,   29,   29,   29, 29,   26,   26,   26,   27,   29,   29,   28,   31},{17
,17,   17,   17,   17,   17,   17,   17,   27,   27,   27,   27,   27,   27,   27,   27,
27,26,   25,25, 25,   23,   23,   23,   22,   22, 22,   22,   22,   22,   21,   21,   21,
19,   25,27,   27,   28,   28,   29,   29,   29,   30,   30,   30,   30,   30,   28,   28,
30,   30,   30,   30,27,   27,   27,   28,   30,   30,   29,   32},{18 ,18,   18,   18,   18,
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
18,   18,   18,   28,   28,28,   28,   28,   27,   27,   27,   27,   27,   26,26, 26,   23,
23,   23,   23,   23,   23,   23,23,   23,   22,   22,   22,   20,   25,   27,   27, 28,
28,   30,   30,   30,   31,   31,   31,   31,31,   28,   28,   31,   31,   31,   31,   27,
27,   27,   28,   30,   30,   30,   33}};

[*Changeover time for a job at location i on machine k*]
RetTrip=
{{11,11,11,11,11,11,11,11,19,19,19,19,19,19,19,19,19,19,18,18,18,16,16,16,16,16,16,16,16,16,1
5,15,15,13,15,16,16,17,17,18,18,18,19,19,19,19,19,17,17,19,19,19,19,16,16,16,17,18,18,18,21},{
12,12,12,12,12,12,12,12,21,21,21,21,21,20,20,20,20,20,19,19,19,17,17,17,17,17,17,17,17,17,16,1
6,16,14,16,17,17,18,18,20,20,20,20,20,20,20,20,18,18,20,20,20,20,17,17,17,18,20,20,19,22},{12,
12,12,12,12,12,12,12,22,22,22,22,22,21,21,21,21,21,20,20,20,18,18,18,17,17,17,17,17,17,17,17,1
7,15,16,18,18,19,19,20,20,20,21,21,21,21,21,19,19,21,21,21,21,18,18,18,19,20,20,20,23},{12,12,
12,12,12,12,12,12,22,22,22,22,22,22,22,22,22,22,20,20,20,19,19,19,18,18,18,18,18,18,17,17,17,1
5,17,19,19,19,19,21,21,21,22,22,22,22,22,19,19,22,22,22,22,19,19,19,19,21,21,20,23}};

[* UNIT LOCATIONS       *]
SupLoc = {3,   3,   3,   3,   2,   2,   2,   2,   2,   2,   2,   2,   2,   2,   2, 2,   2,
   2,   2,   2,   2,   2,   2,   2,   2,   2,   2,   2,   4,   4,   4,   4,   4,   4,
4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,
   4,   4,   4,   4,   4,   4,   4,   4,   1,   1,   1,   1,   1,   1,   1,   1,   1,
1,   1,   1,   1,   1,   1,   1,   1,   1,   4,   4,   4,   4,   4,   4,   4,   4,   4,
   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4,   4};

RD=machCum = {4,   4,   4,   4,   4,   4,   4,   4,   8,   8,   8,   8,   8,   16,
   16,   16,   16,   5,   5,   5,   5,   5,   5,   5,   5,   5,   5,   5,   5,   5,   5,
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
5, 5, 20, 5, 5, 5, 5, 5, 5, 10, 5, 10, 10, 10, 5, 5, 5,
 5, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10};

[*Unit priorities, types and due dates*]
Priority ={5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
 20, 21, 22, 23, 24, 25, 26, 27, 28, 61, 62, 63, 64, 65, 66,
67, 68, 1, 2, 3, 4, 69, 70, 71, 72, 73, 74, 53, 54, 55, 56,
57, 58, 59, 60, 75, 76, 77, 78, 29, 30, 31, 32, 33, 34, 35, 36,
 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52,
 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94,
 95, 96, 97, 98, 99, 100};

Type = {2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
 4, 3, 3, 3, 3, 3, 3, 3, 3, 1, 1, 1, 1, 1, 1, 1, 1,
2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 5, 5, 5, 5, 5, 5, 5, 5, 5,
5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6};
dueDates = {30,40,50,60,80,90};
highWeight=MaxList[Type];
[*Set default weights*]
  weights = Reverse[Table[i,{i,1,highWeight[[1]] }] ]:
ctr=0;
i=k= 0;BESTMACHINE=0;
[* end of data *]
[*--------------------------------------------------------------------------------------------------*]
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
[*Tools*]
[*Machine Schedule Generator*]
CreateSched[ShipSeq]:=Block[{JI=Table[i,{i,NumMachs},{j,1}],k=0,i=0},
          For[k=1,k<=NumJobs,k=k+1,
          ship=ShipSeq[[k]];
          job=Priority[[k]];
          JI[[ship]]=Append[JI[[ship]],job];
          ];
           For[i=1,i<=NumMachs,i=i+1,
           JI[[i]]=Rest[JI[[i]] ];
           ];
JOBINDEX=JI;
];
[*Machine schedule to ship sequence converter*]
[*Converts machine schedule into a ship sequence by job priority*]
Converter[bestSched]:=Block[{job,loc,k,i=0},
 For[k=1,k<=NumMachs,k=k+1,
  For[i=1,i<=JN[[k]],i=i+1,
   job=bestSched[[{k,i}]];
   loc=Search[Priority,job];
   ShipSeq[[loc]]=k;
  ];
 ];
];[*End converter*]

[*Job completion time calculator*]
[*Calculates job completion time list based upon a given ship sequence*]
JCT[ShipSeq]:=Block[{i=0,k,t,num=0,LJ = Table[0,{i,NumMachs}],machCum=RD},
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
 Repeat[
  i=i+1;
  num=Priority[[i]];
  k=ShipSeq[[i]];
    If[ LJ[[k]]<=0,
              t=machCum[[k]]+FirstTrip[[{SupLoc[[num]],k}]]
               +TransT[[{SupLoc[[num]],k}]],
              t=machCum[[k]]+RetTrip[[{SupLoc[[num]],k}]]
               +TransT[[{SupLoc[[num]],k}]]
  ];
  machCum[[k]]=t;
  LJ[[k]]=num;
  JT[[i]]=t;
  ,
  i >= NumJobs
  ];
]; [*End calculator*]
[*--------------------------------------------------------------------------------------------*]
[*Sequential Assignment Heurisitc*]
[*Step 2: Assign Machines to jobs according to preceding order*]
[*Search for the machine which minimizes the job i completion date*]
Repeat[
  i=i+1;
  MINIMUM=9999;
  num=Priority[[i]];
 Repeat[
   k=k+1;
   If[ LJ[[k]]<=0,
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
            t=machCum[[k]]+FirstTrip[[{SupLoc[[num]],k}]]
             +TransT[[{SupLoc[[num]],k}]],
            t=machCum[[k]]+RetTrip[[{SupLoc[[num]],k}]]
             +TransT[[{SupLoc[[num]],k}]]
  ];
   If[t<MINIMUM, MINIMUM=t; BESTMACHINE=k];
   ,
   k>=NumMachs
   ];
[*The BESTMACHINE is assigned to job i*]
  k=0;
  LJ[[BESTMACHINE]]=num;
  machCum[[BESTMACHINE]]=MINIMUM:
  JN[[BESTMACHINE]]=JN[[BESTMACHINE]]+1;
  ShipSeq[[i]]=BESTMACHINE;
  JT[[i]]=MINIMUM;
  Print["mach: ",BESTMACHINE];
  Print["pri: ",i," sel Job: ",num, " on mach: ",BESTMACHINE];
  Print[" num: ",i," Mach Times: ", machCum];


  ,
  i >= NumJobs
  ];
  [* end of Repeat-Job *]


CreateSched[ShipSeq];
bestSeq=ShipSeq;
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
bestSched=JOBINDEX;
bestJT=JT;
oldCum=bestCum=machCum;
OldObj=Sum[weights[[Type[[Priority[[i]] ]] ]]*(Max[0,JT[[i]]-
dueDates[[Type[[Priority[[i]] ]] ]] ]),{i,NumJobs}]:

bestObj=OldObj;
   Print[out1,"bestCum=",bestCum,";",
         "machCum=",machCum,";",
         "bestSched=",bestSched,";",
         "JOBINDEX=",JOBINDEX,";",
         "bestSeq=",bestSeq,";",
         "bestJT=",JT,";",
         "OldObj=",OldObj,";",
         "bestObj=",bestObj,";" ,
         "JN=",JN,";"];

[*------------------------------------------------------------------------------------------------*]

[*Step 3. Improve solution through pairwise exchange*]
SAA[]:=Block[{k=1,i=1,h=1,j=1,L=MaxList[{h,k}] },
  For[k,k<=NumMachs,k=k+1,
  L=MaxList[{h,k}];
   For[i,i<=JN[[k]],i=i+1,
    For[h=L[[1]],h<=NumMachs,h=h+1,
  For[j,j<=JN[[h]],j=j+1,

ctr=ctr+1:
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
      swap=JOBINDEX[[{k,i}]];
      JOBINDEX[[{k,i}]]=JOBINDEX[[{h,j}]];
      JOBINDEX[[{h,j}]]=swap;

[*------------------------------------------------------------------------------------------*]
[*Minimize Weighted Tardiness Objective*]
Converter[JOBINDEX];
JCT[ShipSeq];

NewObj=Sum[weights[[Type[[Priority[[i]] ]] ]]*(Max[0,JT[[i]]-dueDates[[
Type[[Priority[[i]] ]] ]] ]),{i,NumJobs}]:
If[NewObj<OldObj,
            bestCum=machCum;
            machCum=oldCum;
            bestSched=JOBINDEX;
            bestSeq=ShipSeq;
            bestJT=JT;
            OldObj=NewObj;
            bestObj=NewObj,
             JOBINDEX[[{h,j}]]=JOBINDEX[[{k,i}]];
             JOBINDEX[[{k,i}]]=swap;
             machCum=oldCum
];
[*------------------------------------------------------------------------------------------*]
];j=1; [*End For Loop j*]
      ];h=1;L=MaxList[{h,k}];[*End For Loop h*]
    ];i=1; L=MaxList[{h,k}];[*End For Loop i*]
      ];[*End For Loop k*]     ];
```

APPENDIX F

SAA ALGORITHM PROGRAM

```
[*Call Pairwise exchange function*]
SAA[ ];

[*-----------------------------------------------------------------------------------------------------------*]
Print["Best Machine Schedule ",bestSched,"Makespan=",bestCum];
Print["Best Ship sequence=",bestSeq];
Print["Job completion times=",bestJT ];
Print["Best Weighted Tardiness=",bestObj];
```

APPENDIX G

EXHAUSTIVE SEARCH PROGRAM

Exhaustive Search Program For Data Set III

```
RealFormat[10,4];
[*Problem Data*]
NumSup = 4;  [*Number of SPOEs*]
NumDest = 1;  [*Number of SPODs*]
NumJobs = 8;  [*Number of Units*]
NumMachs = 3; [*Number of Ships*]
JT=ShipSeq=Table[0,{i,NumJobs}];
LJ = Table[0,{i,NumMachs}];  [*Initialize Latest Job to Index 0*]
JN = Table[0,{i,NumMachs}];  [*Initialize Machine Job Number to 0*]

[* Time to SPOEs by Ship type for first trip*]
FirstTrip= {{0,12,3},
            {0,12,3},
             {1,11,1},
              {2,11,0}};

[*Processing time for a job at location i on machine k*]
TransT=   {{16,24,28},
           {17,25,29},
            {17,26,30},
             {18,27,31}};
[*Changeover time for a job at locatio i on machine k*]
RetTrip=   {{11,19,19},
            {12,20,20},
             {12,21,21},
              {12,22,22}};
```

APPENDIX G

EXHAUSTIVE SEARCH PROGRAM

```
[*  Unit Locations  *]
SupLoc = {2,1,3,1,4,1,4,4};
[*"Warmup" time for each machine*]
RD=machCum = {4,5,10};
[*Unit priorities, types and due dates*]
Priority ={1,2,3,4,5,6,7,8};
Type = {1,1,2,2,3,3,4,5};
dueDates = {30,40,50,60,80};
highWeight=MaxList[Type];
[*Set default weights*]
  weights = Reverse[Table[i,{i,1,highWeight[[1]] }] ]:
x=Table[0,{i,NumJobs}];
bestSpan =999;
bestSeq = { };
bestObj=OldObj=373;
bestJT=JT;
i=k=0;
ctr=ctr1=ctr2=0;

[* end of data *]
[*-----------------------------------------------------------------------------------------------------*]

[*Function for calculating job completion times*]
JCT[ShipSeq]:=Block[{i=0,k,t,num=0,LJ = Table[0,{i,NumMachs}],machCum=RD},
 Repeat[
  i=i+1;
  num=Priority[[i]];
  k=ShipSeq[[i]];
```

APPENDIX G

EXHAUSTIVE SEARCH PROGRAM

```
  If[ LJ[[k]]<=0,
              t=machCum[[k]]+FirstTrip[[{SupLoc[[num]],k}]]
               +TransT[[{SupLoc[[num]],k}]],
              t=machCum[[k]]+RetTrip[[{SupLoc[[num]],k}]]
               +TransT[[{SupLoc[[num]],k}]]
 ];
 machCum[[k]]=t;
 LJ[[k]]=num;
 JT[[i]]=t;
 ,
 i >= NumJobs
 ];
];
[*End function*]

[*EXHAUSTIVE SEARCH*]

f[n,col]:=Block[{z=1,y=1,v=1,w=1,m=1,k=1,j=1,i=1},
   For[z,z<=n,z=z+1,
   [* Print["z= ",z]; *]
    x[[col-7]]=x[[col-7]]+1;
    If[x[[col-7]]>n,x[[col-7]]=1];
     For[y,y<=n,y=y+1,
    [* Print["y= ",y];*]
     x[[col-6]]=x[[col-6]]+1;
     If[x[[col-6]]>n,x[[col-6]]=1];
     For[v,v<=n,v=v+1,
    [*  Print["v= ",v]; *]
```

APPENDIX G

EXHAUSTIVE SEARCH PROGRAM

```
x[[col-5]]=x[[col-5]]+1;
If[x[[col-5]]>n,x[[col-5]]=1];

    For[w,w<=n,w=w+1,
[*  Print["w= ",w]; *]
    x[[col-4]]=x[[col-4]]+1;
    If[x[[col-4]]>n,x[[col-4]]=1];

     For[m,m<=n,m=m+1,
    [* Print["m= ",m];*]
     x[[col-3]]=x[[col-3]]+1;
     If[x[[col-3]]>n,x[[col-3]]=1];

      For[k,k<=n,k=k+1,
[*   Print["k= ",k];   *]
      x[[col-2]]=x[[col-2]]+1;
      If[x[[col-2]]>n,x[[col-2]]=1];

       For[j,j<=n,j=j+1,
     [*  Print["j= ",j];     *]
       x[[col-1]]=x[[col-1]]+1;
       If[x[[col-1]]>n,x[[col-1]]=1];

        For[i,i<=n,i=i+1,
      [*  Print["i= ",i];*]
        x[[col]]=x[[col]]+1;

ShipSeq=x:
```

APPENDIX G

EXHAUSTIVE SEARCH PROGRAM

```
ctr=ctr+1:
JCT[ShipSeq];

[*Calculate weighted tardiness for generated ship sequence*]

NewObj=Sum[weights[[Type[[Priority[[i]] ]] ]]*(Max[0,JT[[i]]-dueDates[[
Type[[Priority[[i]] ]] ]] ]),{i,NumJobs}]:

If[NewObj<=OldObj,
            bestSeq=ShipSeq;
            bestObj=NewObj;
            OldObj=NewObj;
        bestJT=JT;
            Print[out2,bestObj,bestSeq,JT]
];



              If[x[[col]]==n,x[[col]]=0];
              ];   i=1;
             ];   j=1;
            ];    k=1;
           ];     m=1;
          ];      w=1;
     ];       v=1;
    ];        y=1;
   ];
  ];
```

APPENDIX G

EXHAUSTIVE SEARCH PROGRAM

```
f[3,8]:  [*Call exhaustive search for 3 machines and 8 jobs*]

  Print["counter=",ctr];
  Print["Optimum Ship Sequence is ", bestSeq ];
  Print["Optimum weighted tardiness is ",bestObj];
  Print["Job completion times are ",bestJT];
```

VITA

Name: Garrett Randall Lambert

Occupation: Captain, United States Army
Commissioned May 1984

Education: Bachelor of Science awarded in 1984
United States Military Academy
West Point, NY

Permanent Address: c/o Mr. and Mrs. H. D. Lambert
80 Fairlake
Irvine, CA 92714